



**Desenvolvimento de uma heurística para a determinação de  
rotas de recolha e distribuição de produtos considerando  
múltiplos veículos.**

Daniel Amaro Moura

**Mestrado em Matemática Aplicada à Economia e Gestão**

Trabalho de Projeto orientado por:  
Professora Doutora Maria da Conceição Fonseca



# AGRADECIMENTOS

Agradeço à minha família e aos meus amigos pelo apoio e pela força que me deram neste percurso académico final, especialmente enquanto me preparava para a defesa do trabalho.

Exprimo a minha gratidão à Professora Doutora Maria da Conceição da Fonseca pela orientação, pela disponibilidade para esclarecer dúvidas, pela ajuda essencial que me deu na redação deste projeto e pela leitura crítica e atenta das versões preliminares do trabalho, contribuindo para o seu aperfeiçoamento.

Agradeço à Professora Doutora Ana Maria Duarte Silva Alves Paias e à Professora Doutora Raquel João Fonseca por constituírem o júri, pela leitura e pela análise do projeto e pelas críticas construtivas.



# RESUMO

Neste projeto, apresenta-se um problema de determinação de rotas de recolha e distribuição com escolha seletiva de mercados. Existe uma frota homogénea de veículos e existem pontos onde se faz a recolha de determinados produtos e, posteriormente, faz-se a distribuição pelos clientes, tendo estes uma dada procura que tem que ser satisfeita. Não é obrigatório visitar todos os pontos de recolha disponíveis. O objetivo é construir rotas para os veículos que partam de um depósito, passem por certos mercados para comprar os produtos, entreguem os produtos aos clientes e regressem ao depósito, de modo a minimizar a soma dos custos de aquisição dos produtos e dos custos de deslocação.

Neste trabalho, faz-se uma breve referência a alguns problemas de determinação de rotas para veículos e à sua aplicação a casos reais. Apresenta-se, ainda, um modelo matemático em programação linear inteira mista.

Desenvolve-se, para obter soluções admissíveis para este problema, uma heurística que é constituída por duas fases. A primeira fase consiste em criar rotas com um único cliente e com os mercados que o servem, tendo como base uma heurística desenvolvida para resolver o *Travelling Purchaser Problem*. Nesta primeira fase, constrói-se uma solução inicial, a qual é melhorada através de dois procedimentos: *Market drop* e *Market exchange*. A segunda fase consiste na fusão das rotas obtidas, juntando vários clientes na mesma rota, de modo a diminuir os custos de viagem.

Os resultados computacionais são obtidos para dados gerados aleatoriamente, considerando duas áreas onde estão os clientes, o depósito e os mercados, dois tipos de probabilidade associados à existência de determinado produto em cada mercado e à probabilidade de a procura de determinado produto por parte de um cliente ser superior a zero e diferentes valores para número de mercados e procura. Faz-se uma análise dos resultados obtidos em termos de média das melhorias percentuais quando se faz a fusão de rotas e em termos de tempos computacionais, considerando duas capacidades diferentes para os veículos.

Palavras-chave: *Travelling Purchaser Problem*, rotas de recolha e distribuição de produtos, programação linear inteira mista, heurística.



# ABSTRACT

In this project, a pickup and delivery problem with selective choice of markets is presented. There is a fleet of homogenous vehicles which travels through pickup points to get certain products and then delivers them to the customers who have a certain demand that must be satisfied. It is not necessary to visit every available pickup point. The goal is to find a good, next to optimal, route for the vehicles that leave the depot, stop at certain markets where products are bought, deliver those products to the customers and then return to the depot, in order to minimize the sum of the purchasing costs and the travelling costs.

In this project, a brief reference to some vehicle routing problems and some of its applications to the real world is made. A mixed integer linear programming model is presented.

A heuristic is built to find feasible solutions for this problem. The heuristic consists of two phases, the first of which, consists of creating routes with a single customer and the markets which satisfy the customer's demand. This phase is based on a heuristic for the Travelling Purchaser Problem where an initial feasible solution is found and improved upon through two procedures: Market Drop and Market Exchange. The second phase consists of merging the routes obtained beforehand, joining multiple customers in the same route, in order to decrease travelling costs.

Some computational results were obtained for randomly generated data, considering two different areas for the depot, customers and markets, two different probabilities for the existence of a certain product in a certain market, two different probabilities for the existence of demand of a certain product for each customer and, lastly, different numbers of customers and markets. The results were analysed in regards to the average percentage improvements for the route merging, as well as regarding the computational time, considering two different maximum vehicle capacities.

Keywords: Travelling Purchaser Problem, pickup and delivery routes, mixed integer linear programming, heuristic.





# ÍNDICE

Agradecimentos.....	i
Resumo.....	iii
Abstract .....	v
Índice.....	vii
Lista de tabelas .....	ix
Lista de figuras .....	xi
Lista de algoritmos .....	xiii
1. Introdução.....	1
2. Modelo Matemático .....	3
3. Heurística para o problema.....	7
3.1. Obtenção de rotas com um só cliente.....	8
3.2. Obtenção de rotas com vários clientes .....	10
3.3. Descrição detalhada do algoritmo .....	11
3.3.1. Algoritmo para obter rotas com um único cliente .....	12
3.3.2. Algoritmo para a fusão de rotas.....	14
3.4. Fluxogramas .....	17
4. Resultados computacionais .....	19
4.1. Exemplo da formação de rotas .....	21
4.2. Apresentação e análise dos resultados.....	24
5. Conclusões .....	27
Referências .....	29
Anexo .....	31



# LISTA DE TABELAS

Tabela 4.1: Grupos de dados gerados.....	20
Tabela 4.2: Comparação dos custos finais .....	25
Tabela 4.3: Média das melhorias por probabilidade da procura.....	25
Tabela 4.4: Média das melhorias por áreas .....	25
Tabela 4.5: Média das melhorias por par cliente/mercado .....	26
Tabela 4.6: Comparação dos tempos computacionais.....	26
Tabela A.1: Resultados finais dos Grupos 1 e 2.....	31
Tabela A.2: Resultados finais dos Grupos 3 e 4.....	32



# LISTA DE FIGURAS

Figura 3.1: Exemplo de rotas sem mercados em comum. ....	15
Figura 3.2: Fusão de rotas sem mercados em comum. ....	15
Figura 3.3: Exemplo de rotas com mercados em comum.....	16
Figura 3.4: Fusão de rotas com mercados em comum .....	16
Figura 3.5: Fluxograma da função $\text{custorota}$ .....	17
Figura 3.6: Fluxograma da função $\text{custocompras}$ .....	18
Figura 4.1: Rota $O, M1, C1, O$ .....	22
Figura 4.2: Rota $O, M7, C2, O$ .....	22
Figura 4.3: Rota $O, M7, M4, C3, O$ .....	23
Figura 4.4: Rota $O, M9, M1, C4, O$ .....	23
Figura 4.5: Rota $(O, M7, M8, M1, C5, O)$ .....	23
Figura 4.6: Rota $(O, M7, C1, C5, M8, C4, M4, C3, M1, C2, O)$ .....	24



# LISTA DE ALGORITMOS

Algoritmo 3.1 .....	12
Algoritmo 3.2 .....	14





# 1. INTRODUÇÃO

O problema de determinação de rotas para um ou vários veículos é um problema de grande aplicação na vida real, nomeadamente na área da logística. Um dos problemas desta área mais simples e mais estudados é o problema de determinação de rotas para vários veículos (*Vehicle Routing Problem* - VRP), considerando uma capacidade limitada nos veículos (*Capacitated Vehicle Routing Problem* - CVRP).

Existem variantes deste problema. Em alguns problemas de determinação de rotas de veículos, considera-se, associado a cada cliente, uma janela temporal na qual o serviço ao cliente deve começar. Neste caso, pode-se considerar ou que estas restrições podem ser violadas incorrendo numa penalização ou que têm de obrigatoriamente ser cumpridas, não sendo permitido a um veículo chegar nem antes nem depois do tempo correspondente à janela temporal. Encontram-se, neste caso, entre outros, problemas relacionados com a entrega de dinheiro nos bancos, recolha nos correios e rotas dos autocarros escolares (ver [1]).

Uma importante família de problemas de determinação de rotas para vários veículos está relacionada com o transporte de mercadoria ou passageiros entre pontos origem e pontos destino. Este problema designa-se por problema de determinação de rotas com recolha e distribuição (*Pickup and Delivery Vehicle Routing Problem* - PDVRP). Algumas aplicações são a gestão das bicicletas em vários pontos num sistema de partilha e sistemas de *car sharing*.

Dentro desta família de problemas, tem-se a de recolha e distribuição simultânea, em que um ponto de distribuição é também potencialmente um ponto de recolha. Algumas aplicações são, por exemplo, a recolha e distribuição de medicamentos nas farmácias e de bebidas em bares ou cafés, em que, quando se efetua a entrega de medicamentos e bebidas, recolhem-se medicamentos fora de prazo ou devolvidos e garrafas vazias de bebidas, respetivamente (ver [2]).

Note-se que há exemplos pertencentes a várias famílias de problemas de determinação de rotas para veículos.

Os problemas de aprovisionamento e transporte, em geral, consistem em planear quantos produtos se comprem e onde os comprar para satisfazer uma procura de modo a minimizar os custos de compras, viagens e armazenamento. Há interesse nestes problemas, visto que, um fator que aumenta o preço final de um produto é o custo do seu transporte. (ver [3]).

Ting *et al.*, em [4], apresentam uma aplicação relacionada com o serviço de partilha de bicicletas. Visitar só alguns dos pontos de partilha para recolher as bicicletas e transportá-las para os locais onde há maior procura nas horas de ponta tem menor custo do que visitar todos os pontos de partilha. Estes autores apresentam três meta-heurísticas para o problema: *tabu search*, algoritmo genético e *scatter search*.

Contrariamente ao que acontece com as variantes do VRP apresentadas anteriormente, existe uma variante em que o conjunto dos clientes a visitar não é dado *a priori*. Em geral, existe um conjunto de clientes com uma determinada procura e um lucro associado a cada cliente. Não sendo possível ou necessário satisfazer a procura de todos os clientes, por exemplo, por haver uma restrição de uma janela temporal, tem que se decidir quais os clientes a servir e em que rotas os incluir, com o objetivo de maximizar o lucro (ver [5]).

Devido à complexidade computacional destes problemas, há um foco em tentar construir heurísticas que resultem numa boa, ainda que não ótima, solução admissível que vise minimizar os custos ou maximizar os lucros, conforme o seu objetivo.

Neste projeto, desenvolve-se e implementa-se uma heurística para resolver um problema de determinação de rotas de recolha e distribuição de produtos com escolha seletiva de mercados. Considera-se que existe uma frota de veículos homogêneos e que os produtos estão disponíveis em certos mercados

e que cada mercado pratica um preço distinto. Os veículos partem de um depósito, adquirem os produtos em alguns dos mercados em que estes estão disponíveis e fazem a sua distribuição pelos clientes, garantindo que a procura destes fica completamente satisfeita. Considera-se uma capacidade máxima para os veículos e não é obrigatório visitar todos os mercados. A heurística desenvolvida neste trabalho de projeto, dadas a procura dos clientes, as matrizes de custos de viagem entre cada ponto de interesse, os custos unitários e as quantidades disponíveis de cada produto em cada mercado e a capacidade máxima dos veículos, procura uma solução admissível que visa minimizar os custos totais, ou seja, a soma dos custos de compras e dos custos de viagem.

A heurística é constituída por uma fase de construção de uma solução admissível constituída por um número de rotas igual ao número de clientes. Cada uma destas rotas inclui o depósito, os mercados que servem o cliente e o cliente cuja procura é satisfeita. Esta fase inclui, ainda, melhoramentos locais, através dos procedimentos *Market Drop* e *Market Exchange*. Esta fase de construção da heurística é baseada em heurísticas existentes para o *Travelling Purchaser Problem*. Segue-se uma fase de junção de rotas.

Este problema já foi introduzido por Palomo-Martínez e Salazar-Aguilar, em [6]. Estes autores consideram dois objetivos: minimizar o custo total e minimizar o tempo de espera dos clientes. Estes autores referem uma aplicação deste problema: uma equipa de técnicos de informática e telecomunicação de uma empresa parte do seu escritório, vai buscar as ferramentas de que precisa e presta o seu serviço a outros escritórios subsidiários da empresa.

Este trabalho de projeto está estruturado do seguinte modo: no capítulo 2, apresenta-se um modelo em programação linear inteira mista para o problema; no capítulo 3, apresenta-se a heurística desenvolvida para o problema; no capítulo 4, apresentam-se os resultados computacionais obtidos para dados gerados aleatoriamente e, finalmente, no capítulo 5, apresentam-se as conclusões.

## 2. MODELO MATEMÁTICO

Neste capítulo, apresenta-se um modelo em programação linear inteira mista para o problema em estudo de determinação de rotas de recolha e distribuição de produtos com escolha seletiva de mercados.

O modelo em programação linear inteira apresentado para o problema de determinação de rotas de recolha e distribuição de produtos com escolha seletiva de mercados baseia-se no modelo proposto por Salvendy e Sol, em [7], para o problema de determinação de rotas de recolha e distribuição de produtos com janelas temporais e no modelo apresentado por Oliveira, em [8], para um problema de planeamento da distribuição de produtos agrícolas num canal curto de distribuição.

Algumas hipóteses:

- Os veículos estão disponíveis num depósito e iniciam e terminam a rota nesse depósito.
- Em cada rota, um mercado é visitado, no máximo, uma vez.
- O tempo total da rota está de acordo com as restrições relativas ao horário dos motoristas dos veículos.
- A procura dos clientes deve ser satisfeita.
- A disponibilidade de cada produto, em cada mercado, não pode ser excedida.
- Não é obrigatório visitar todos os mercados.
- A frota de veículos é homogénea.

Notação usada:

- $P$  - Conjunto de produtos
- $Cl$  - Conjunto de clientes
- $M$  - Conjunto de mercados
- $q_{nh}$  - Procura do produto  $n \in P$  relativa ao cliente  $h \in Cl$
- $a_{nj}$  - Quantidade de produto  $n \in P$  disponível no mercado  $j \in M$
- $pa_{nj}$  - Custo unitário do produto  $n \in P$  no mercado  $j \in M$
- $VH$  - Conjunto de veículos
- $Q$  - Capacidade comum a todos os veículos  $v \in VH$

Considere-se o grafo  $G = (X, A)$  em que  $X = Cl \cup M \cup \{O\}$ . Consideram-se as réplicas  $O^+$  e  $O^-$  do depósito que representam o depósito, respetivamente, à partida e à chegada dos veículos.  $A = \{(i, j): i, j \in X\}$ .

Como o número de veículos é estabelecido *a priori*, podem não ser todos usados. Assim, considera-se um arco entre a réplica do depósito que o representa à partida,  $O^+$ , e a réplica do depósito que o representa à chegada,  $O^-$ . Assim,  $A \leftarrow A \cup \{(O^+, O^-)\}$ .

Alguns dos arcos de  $G$  podem ser eliminados do conjunto  $A$ .

- Os veículos saem do depósito e os primeiros locais que visitam são mercados. Assim, só se consideram em  $A$  os arcos  $\{(O^+, j): j \in M\}$  e o arco  $(O^+, O^-)$ , no caso do veículo não ser utilizado.
- De modo idêntico a chegada ao depósito é feita sempre a partir de clientes. Assim, só se consideram em  $A$  os arcos  $\{(k, O^-): k \in Cl\}$  e o arco  $(O^+, O^-)$ , no caso do veículo não ser utilizado.
- Não existem loops. Os arcos  $(i, j)$  com  $i = j$  não pertencem a  $A$ .
- $ct_{ij}$  representa a distância entre  $i$  e  $j$ , com  $(i, j) \in A$ .

Variáveis de decisão:

- $z_h^v = \begin{cases} 1 & \text{se o cliente } h \in Cl \text{ é visitado pelo veículo } v \in VH \\ 0 & \text{caso contrário} \end{cases}$
- $g_{jh}^v = \begin{cases} 1 & \text{se o mercado } j \in M \text{ e o cliente } h \in Cl \text{ pertencem à rota do veículo } v \in VH \\ 0 & \text{caso contrário} \end{cases}$
- $x_{ls}^v = \begin{cases} 1 & \text{se o veículo } v \in VH \text{ atravessa o arco } (l, s) \in A \\ 0 & \text{caso contrário} \end{cases}$
- $f_{njh}$  - Quantidade de produto  $n \in P$  comprado no mercado  $j \in M$  para o cliente  $h \in Cl$
- $y_l$  - Carga do veículo quando chega a  $l \in X \cup O$

Como o número de veículos é estabelecido *a priori*, tem-se  $x_{o^+o^-}^v \neq 0$  para algum veículo,  $v$ , não usado. Sendo uma frota homogênea em que a capacidade de cada veículo é a mesma e igual a  $Q$ , tem-se (ver [9]) que o número mínimo de veículos é  $\#(VH)$  que verifica:

$$\frac{\sum_{n \in P, h \in Cl} q_{nh}}{Q} \leq \#(VH).$$

O modelo em Programação linear inteira mista é o seguinte:

$$\min \sum_{v \in VH} \sum_{(l,s) \in A} ct_{ls} x_{ls}^v + \sum_{j \in M} \sum_{k \in Cl} \sum_{p \in P} p a_{pj} f_{pjk} \quad (2.1)$$

s.a.

$$\sum_{v \in VH} z_h^v = 1 \quad \forall h \in Cl \quad (2.2)$$

$$g_{jh}^v \leq z_h^v \quad \forall j \in M, h \in Cl, v \in VH \quad (2.3)$$

$$\sum_{s: (j,s) \in A} x_{js}^v = g_{jk}^v \quad \forall j \in M, k \in Cl, v \in VH \quad (2.4)$$

$$\sum_{s: (k,s) \in A} x_{ks}^v = z_k^v \quad \forall k \in Cl, v \in VH \quad (2.5)$$

$$\sum_{s: (l,s) \in A} x_{ls}^v - \sum_{s: (s,l) \in A} x_{sl}^v = 0 \quad \forall l \in X, v \in VH \quad (2.6)$$

$$\sum_{s \in M \cup O^-} x_{o^+s}^v = 1 \quad \forall v \in VH \quad (2.7)$$

$$\sum_{s \in Cl \cup O^+} x_{so^-}^v = 1 \quad \forall v \in VH \quad (2.8)$$

$$f_{njh} \leq \bar{U} (g_{jh}^v + 1 - z_h^v) \quad \forall n \in P, j \in M, h \in Cl, v \in VH \quad (2.9)$$

$$\sum_{h \in Cl} f_{njh} \leq a_{nj} \quad \forall n \in P, j \in M \quad (2.10)$$

$$\sum_{j \in M} f_{njh} = q_{nh} \quad \forall n \in P, h \in Cl \quad (2.11)$$

$$y_{O^+} = 0 \quad (2.12)$$

$$y_{O^-} = 0 \quad (2.13)$$

$$y_j + \sum_{h \in Cl} \sum_{n \in P} f_{njh} \leq y_s + Q(1 - x_{js}^v) \quad \forall v \in VH, j \in M, s: (j, s) \in A \quad (2.14)$$

$$y_h - \sum_{j \in M} \sum_{n \in P} f_{njh} \leq y_s + Q(1 - x_{hs}^v) \quad \forall v \in VH, h \in Cl, s: (h, s) \in A \quad (2.15)$$

$$\sum_{v \in VH} Qz_h^v \geq y_h \quad \forall h \in Cl \quad (2.16)$$

$$\sum_{v \in VH} Qg_{jh}^v \geq y_j \quad \forall h \in Cl, j \in M \quad (2.17)$$

$$\text{Eliminação de subcircuitos} \quad (2.18)$$

$$z_h^v \in \{0, 1\} \quad \forall h \in Cl, v \in VH \quad (2.19)$$

$$g_{jh}^v \in \{0, 1\} \quad \forall j \in M, h \in Cl, v \in VH \quad (2.20)$$

$$x_{ls}^v \in \{0, 1\} \quad \forall (l, s) \in A, v \in VH \quad (2.21)$$

$$f_{njh} \geq 0 \quad \forall n \in P, j \in M, h \in Cl \quad (2.22)$$

$$y_l \geq 0 \quad \forall l \in X \quad (2.23)$$

onde  $\bar{U}$  é uma constante suficientemente grande.

A expressão (2.1) é a minimização da função objetivo que representa os custos totais envolvidos: custos de transporte e custos de aquisição dos produtos.

As restrições (2.2) garantem que cada cliente é visitado por um único veículo. As restrições (2.3) garantem que, se um veículo não visita um cliente, então não pode visitar os mercados que lhe estão associados. As restrições (2.4) e (2.5) garantem que, se um mercado ou um cliente, respetivamente, é visitado por um veículo, então este sai do mercado ou do cliente, respetivamente, apenas uma vez. As restrições (2.6) são restrições de conservação do fluxo, isto é, se um veículo chega a um local, então também tem de sair dele. As restrições (2.7) e (2.8) garantem que cada veículo inicia e termina a viagem no depósito. Depois de sair do depósito, cada veículo visita um mercado e chega ao depósito depois de visitar um cliente ou vai diretamente do depósito origem para o depósito destino se não for usado. As restrições (2.9) garantem que, se um veículo visita um cliente e não visita determinado mercado, então este mercado não fornece qualquer produto para esse cliente. As restrições (2.10) garantem que a disponibilidade de produtos nos mercados não é excedida e as restrições (2.11) garantem que a procura de cada cliente é satisfeita. As restrições (2.12) e (2.13) garantem que o veículo parte e, respetivamente, chega ao depósito vazio. As restrições (2.14) e (2.15) dizem respeito à carga do veículo quando chega a cada local. As restrições (2.16) e (2.17) garantem que a capacidade de cada veículo não é excedida. As restrições (2.19) até (2.23) definem o domínio das variáveis.

As restrições (2.18) dizem respeito à eliminação de subcircuitos. Existem muitas versões polinomiais para este tipo de restrições. Oncan, Altimel e Laporte apresentam, em [10], várias versões para

estas restrições no caso do problema do caixeiro-viajante e fazem uma análise das formulações resultantes. Neste artigo, referem as restrições apresentadas por Sarin *et al.*, em [11], que são usadas por Rais *et al.*, em [12], para o problema de determinação de rotas de recolha e distribuição com pontos de transfeira. Estas são as desigualdades usadas neste modelo para garantir a eliminação de subcircuitos.

Definem-se as variáveis para  $i, j \in X \cup \{O\}$  e  $v \in VH$ :

$$w_{ij}^v = \begin{cases} 1 & \text{se o vértice } i \text{ precede o vértice } j \text{ na rota do veículo } v \\ 0 & \text{caso contrário} \end{cases}$$

Tem-se:

$$\bullet \quad \forall i, j \in X \setminus \{O\} \quad \forall v \in VH: \quad x_{ij}^v \leq w_{ij}^v. \quad (2.24)$$

$$\bullet \quad \forall i, j \in X \setminus \{O\} \quad \forall v \in VH: \quad w_{ij}^v + w_{ji}^v = 1. \quad (2.25)$$

$$\bullet \quad \forall i, j, l \in X \setminus \{O\} \quad \forall v \in VH: \quad w_{ij}^v + w_{jl}^v + w_{li}^v \leq 2. \quad (2.26)$$

Este problema é de grande complexidade pelo que, embora fosse interessante calcular o desvio entre o limite superior obtido pela heurística e o valor ótimo, optou-se por apresentar apenas o modelo neste trabalho de projeto.

### 3. HEURÍSTICA PARA O PROBLEMA

Neste capítulo, apresenta-se a heurística desenvolvida para o problema de determinação de rotas de recolha e distribuição de produtos com escolha seletiva de mercados.

Esta heurística consiste em duas fases. Numa primeira fase, obtém-se uma rota inicial só com um cliente e os mercados que o servem e, numa segunda fase, faz-se a fusão de rotas desde que não se exceda a capacidade dos veículos.

Para cada cliente, numa fase de construção, elabora-se uma rota inicial e, posteriormente, numa fase de melhoramento, a rota inicial é melhorada, obtendo-se uma rota individual. Depois de obtidas as rotas individuais, há uma fase de fusão das rotas.

Na fase de construção, escolhido aleatoriamente um cliente, decide-se, iterativamente, quais os mercados a visitar e a ordem por que estes são visitados de modo a minimizar os custos totais, isto é, custos de viagem mais custos de aquisição dos produtos. Nos mercados visitados, para servir este cliente, compra-se cada produto procurado onde este é mais barato, de modo a minimizar os custos das compras.

Na fase de melhoramento, tenta-se melhorar a rota inicial através da remoção e da substituição de mercados visitados. Para cada mercado visitado, verifica-se se é possível servir o cliente sem visitar esse mercado. Se for possível e se o custo de servir o cliente diminuir em relação ao custo da rota em que se visita o mercado, então este é removido da rota (*Market Drop*). Na sequência deste procedimento, para cada par de mercados visitado-não visitado, verifica-se se é possível servir o cliente substituindo o visitado pelo não visitado e, se for possível servir o cliente e se o custo diminuir, então substitui-se o mercado visitado pelo não visitado na rota (*Market Exchange*).

Na fase de fusão das rotas, visitam-se mercados e serve-se um cliente e, em vez de voltar logo para o depósito, visita-se outro conjunto de mercados e serve-se outro cliente. Para respeitar a restrição de, no máximo, só visitar cada mercado uma vez, se dois clientes são servidos pelo mesmo mercado, só se juntam os clientes na mesma rota, nesta fase, se não se ultrapassar a capacidade máxima dos veículos. Quando o mercado comum é visitado pela primeira vez, compram-se os produtos para servir os dois clientes e o mercado não é visitado uma segunda vez. Faz-se iterativamente a junção de clientes na mesma rota, enquanto a capacidade do veículo não for excedida. Quando não for possível continuar com a junção na mesma rota, juntam-se iterativamente os restantes clientes noutra rota do mesmo modo e repete-se o processo.

O algoritmo para a obtenção das rotas para o problema em estudo tem o seguinte esquema:

**INPUT:** Matrizes das distâncias entre mercados, clientes e depósito, procura dos clientes para cada produto, disponibilidade de cada produto em cada mercado, preços de aquisição dos produtos em cada mercado, capacidade dos veículos.

Algoritmo 1: Construção de rotas com um só cliente e melhoramento dessas rotas.

Algoritmo 2: Fusão das rotas obtidas anteriormente.

**OUTPUT:** Rotas finais, compras feitas em cada mercado, em cada rota e custo final que inclui o custo associado às viagens e às compras.

### 3.1. OBTENÇÃO DE ROTAS COM UM SÓ CLIENTE

A heurística usada para obter as rotas individuais é baseada nas heurísticas usadas por Boctor, Laporte e Renaud, em [13], para a resolução do *Travelling Purchaser Problem* (TPP). O TPP é um problema NP-difícil que consiste na determinação de uma rota para um veículo que sai de um armazém ou depósito e compra um conjunto de produtos em vários mercados e volta ao local de partida. Os mercados praticam preços distintos e não é obrigatório visitá-los a todos. Pode considerar-se que cada mercado tem quantidade suficiente para satisfazer a procura de cada produto que vende ou que cada mercado pode não ter quantidade suficiente. Pode considerar-se, ainda, que as distâncias são simétricas ou assimétricas.

Este problema foi inicialmente proposto por Ramesh em [14]. Este autor desenvolveu dois algoritmos para este problema, considerando a versão em que cada mercado tem quantidade suficiente para satisfazer a procura dos produtos que vende.

Uma aplicação do TPP é, por exemplo, o escalonamento de  $n$  processos numa máquina com  $m$  estados. Há um custo associado a mudar o estado da máquina e um custo associado a cada processo em cada estado da máquina. O objetivo é minimizar a soma dos custos. Esta aplicação é apresentada por Ong, em [15].

Singh e Oudheusden, em [16], apresentam um algoritmo de *branch and bound* para o TPP. Os limites inferiores para o valor ótimo do problema são obtidos relaxando algumas das restrições do TPP, o que resulta num problema de localização simples. Os resultados obtidos mostram que se conseguem resolver problemas com até 25 mercados e 100 produtos em tempo computacional razoável. Apresentam ainda algumas aplicações do TPP nas áreas do sequenciamento de tarefas, recolha de produtos em armazéns depois de receber o pedido de um cliente e rotas de autocarros de escolas.

Laporte *et al.*, em [17], apresentam um algoritmo *branch and cut* para a resolução do TPP em que as distâncias são simétricas e para ambos os casos em que a disponibilidade dos mercados é ou não limitada. Estes autores apresentam um modelo em programação linear inteira para o problema e várias desigualdades válidas. Testaram o algoritmo para quatro classes de exemplos gerados aleatoriamente. Referem que os resultados obtidos são melhores do que os apresentados por Singh e Oudheusden em [16] para o caso em que as disponibilidades dos mercados são ilimitadas. Referem ainda que desconhecem a existência de outros estudos até à data de publicação do seu artigo para a variante do TPP em que a disponibilidade nos mercados é limitada.

Choi e Lee, em [18], apresentam outras versões do problema. Neste artigo, os autores referem ainda alguns dos trabalhos mais importantes apresentados para cada versão apresentada do TPP.

Bernardino e Paias, em [19], apresentam meta-heurísticas para o problema que combinam algoritmos genéticos com a pesquisa local e fizeram experiências computacionais para resolver instâncias de grande dimensão em tempos de computação razoáveis.

O problema que se pretende resolver, neste projeto, para obter rotas para cada um dos veículos é diferente, visto que, em vez de voltar para o depósito com os produtos comprados nos vários mercados, o veículo distribui os produtos pelo cliente e só depois volta ao depósito. Cada cliente tem um conjunto de produtos que pretende adquirir e em determinadas quantidades. Os produtos podem ser adquiridos em vários mercados. Os preços praticados e as quantidades disponíveis em cada mercado variam. Cada mercado só pode ser visitado, no máximo, uma vez por rota. Existe um limite à capacidade do veículo. Cada cliente pertence a uma e só uma rota e a capacidade máxima do veículo não pode ser excedida em cada rota.



Pretende-se determinar as rotas, garantindo a satisfação da procura dos clientes, de modo a minimizar os custos globais. Estes custos englobam os custos de aquisição dos produtos e os custos de deslocação. Cada rota começa e termina no depósito onde está o veículo e é constituída pelos mercados onde são comprados os produtos e o cliente a quem se destinam.

Note-se que se se considerar que o único cliente é o depósito, o problema transforma-se numa versão do *Travelling Purchaser Problem*.

Apresenta-se, seguidamente, a heurística desenvolvida, neste projeto, para o problema de determinação de rotas de recolha e distribuição de produtos para um único cliente com escolha seletiva de mercados.

Fase de construção:

- Escolhe-se aleatoriamente um cliente.
- Reordena-se o conjunto de produtos.
- O primeiro mercado a visitar vai ser o que minimiza a expressão:

$$2 \frac{\text{custo de viagem depósito/mercado}}{\text{quantidade de produto disponível}} + \text{custo do unitário do produto}.$$

- Enquanto a procura do cliente não estiver satisfeita, escolhe-se um mercado novo para visitar. O mercado a escolher será o que minimiza a soma entre o custo das viagens entre depósito, mercados e cliente e o custo das compras dos produtos nos mercados escolhidos. Quando se adiciona um mercado à rota, insere-se numa posição de acordo com o vizinho mais próximo. Nos mercados visitados, compram-se os produtos onde estes são mais baratos sem ultrapassar a procura.

Na fase de melhoramento local, consideram-se dois tipos de operações: *market drop*, que consiste em retirar da rota um mercado, se se obtiver uma melhor solução, e *market exchange*, que consiste em substituir um mercado da rota por outro se se obtiver uma melhor solução.

Fase de melhoramento 1 (*Market drop*):

- Escolhe-se aleatoriamente um mercado visitado.
- Retira-se o mercado se comprando os produtos relativos a este mercado nos outros mercados visitados resultar num menor custo total.
- Repetir até não haver melhoramentos possíveis.

Fase de melhoramento 2 (*Market exchange*):

- Escolhe-se aleatoriamente um mercado visitado e um mercado não visitado.
- Retira-se o mercado visitado e adiciona-se o mercado não visitado se resultar num menor custo total.
- Repetir até não haver melhoramentos possíveis.

Voltar à fase de construção e passar ao próximo cliente, se a procura de algum não estiver satisfeita.

## 3.2. OBTENÇÃO DE ROTAS COM VÁRIOS CLIENTES

Depois de obter as rotas individuais constituídas, cada uma, por um cliente e os mercados que o servem, faz-se uma fusão das rotas para obter rotas com vários clientes.

Fase de junção de rotas:

- Escolhe-se aleatoriamente um cliente do conjunto de clientes, cada um dos quais faz parte de uma rota individual.
- Juntam-se iterativamente os outros clientes e os respetivos mercados associados à rota do cliente escolhido, caso não tenham nenhum mercado em comum com os clientes da rota.
- Caso hajam mercados em comum, se a capacidade máxima de carga não é ultrapassada, junta-se o cliente à rota e compram-se as quantidades correspondentes à procura desse cliente nesses mercados quando estes são visitados.
- Quando não se puderem juntar mais clientes nesta rota, se o conjunto de clientes com rotas individuais tiver dois ou mais elementos, voltar ao início desta fase.

Obtêm-se, deste modo, as rotas de recolha e distribuição de produtos com escolha seletiva de mercados.

Na próxima secção, apresenta-se uma descrição detalhada da heurística usada.

### 3.3. DESCRIÇÃO DETALHADA DO ALGORITMO

Sejam  $Cl$  o conjunto de clientes, com  $\#Cl = k$ ,  $M$  o conjunto de mercados, com  $\#M = m$ , e  $O$  o depósito. Seja  $P$  o conjunto de produtos, com  $\#P = p$ .

Para cada cliente  $j \in Cl$ , a procura relativa ao produto  $i \in P$  é  $q_{ij}$ .

A quantidade disponível do produto  $i \in P$  no mercado  $j \in M$  é  $a_{ij}$  e o preço de aquisição unitário é dado por  $pa_{ij}$ .

A capacidade máxima de cada veículo é  $Q$ .

$[c]_{ij}$  é a entrada  $(i, j)$  da matriz de custos associados a viajar de  $i$  para  $j$ , com  $i, j \in M \cup \{O\}$ , ou seja, custos associados a viajar entre mercados/depósito e mercados/depósito.

$$\begin{matrix} & M_1 & \cdots & M_m & O \\ \begin{matrix} M_1 \\ \vdots \\ M_m \\ O \end{matrix} & \begin{pmatrix} c_{1,1} & \cdots & c_{1,m} & c_{1,m+1} \\ \vdots & \ddots & \vdots & \vdots \\ c_{m,1} & \cdots & c_{m,m} & c_{m,m+1} \\ c_{m+1,1} & \cdots & c_{m+1,m} & c_{m+1,m+1} \end{pmatrix} \end{matrix}$$

$[c']_{ij}$  é a entrada  $(i, j)$  da matriz de custos associados a viajar de  $i$  para  $j$ , com  $i \in M \cup \{O\}, j \in Cl$ , ou seja, custos associados a viajar entre mercados e depósito e clientes.

$$\begin{matrix} & Cl_1 & \cdots & Cl_k \\ \begin{matrix} M_1 \\ \vdots \\ M_m \\ O \end{matrix} & \begin{pmatrix} c'_{1,1} & \cdots & c'_{1,k} \\ \vdots & \ddots & \vdots \\ c'_{m,1} & \cdots & c'_{m,k} \\ c'_{m+1,1} & \cdots & c'_{m+1,k} \end{pmatrix} \end{matrix}$$

$[c'']_{ij}$  é a entrada  $(i, j)$  da matriz de custos associados a viajar de  $i$  para  $j$ , com  $i \in Cl, j \in Cl$ , ou seja, custos associados a viajar entre clientes.

$$\begin{matrix} & Cl_1 & \cdots & Cl_k \\ \begin{matrix} Cl_1 \\ \vdots \\ Cl_k \end{matrix} & \begin{pmatrix} c''_{1,1} & \cdots & c''_{1,k} \\ \vdots & \ddots & \vdots \\ c''_{k,1} & \cdots & c''_{k,k} \end{pmatrix} \end{matrix}$$

Assume-se que as matrizes de custos são simétricas. Assume-se, também, que se verifica a desigualdade triangular nos custos, ou seja, o custo de viajar de A para B é menor ou igual ao custo de viajar de A para C mais o custo de viajar de C para B.

### 3.3.1. ALGORITMO PARA OBTER ROTAS COM UM ÚNICO CLIENTE

No algoritmo 3.1, apresenta-se, de forma detalhada, o método descrito anteriormente para obtenção de rotas com um só cliente.

#### Algoritmo 3.1

##### Fase de construção

##### Passo 1:

Reordenar o conjunto  $Cl = \{1, \dots, k\}$  aleatoriamente. Seja  $f_{i,j,x} := 0$  a quantidade de produto  $i$  comprada no mercado  $j$  para o cliente  $x$ . Seja  $x := 1$  (para simplificação de notação,  $x$  representa o elemento  $x$  do conjunto  $Cl$ ).

(Repetir até  $x = k$  inclusivé)

##### Passo 2:

Sejam  $MNV := \{1, \dots, m\}$  o conjunto de mercados não visitados e  $MV := \emptyset$  o conjunto de mercados visitados na rota relativa ao cliente  $x$ .

Reordenar o conjunto de produtos em que a procura do cliente  $x$  é maior que zero,  $P_x = \{1, \dots, p_x\}$ , aleatoriamente. Seja  $y := 1$  (para simplificação de notação  $y$  representa o elemento  $y$  do conjunto  $P_x$ ).

##### Passo 3:

Determinar o mercado  $i^*$  que minimize a expressão.  $j^* = \arg \min_{j \in MNV} \left\{ 2 \frac{c_{m+1,i}}{a_{1,j}} + pa_{1,j} \right\}$ .

Comprar o máximo de cada produto sem ultrapassar a procura,  $f_{i,j^*,x} \leftarrow \min\{q_{i,x}, a_{i,j^*}\}$  para  $i \in P$ .

Define-se a rota parcial:  $T_x := \{O, j^*\}$ .

Retira-se o mercado já visitado do conjunto de mercados por visitar:  $MNV \leftarrow MNV \setminus \{j^*\}$ .

Adiciona-se o mercado visitado ao conjunto de mercados visitados:  $MV \leftarrow MV \cup \{j^*\}$ .

##### Passo 4:

(Repetir até  $\sum_{j \in M} f_{i,j,x} = q_{i,x} \forall i \in P_x$ ) até toda a procura do cliente  $x$  ser satisfeita

(Repetir até  $\sum_{j \in M} f_{y,j,x} = q_{y,x}$ ) até a procura do cliente  $x$  para o produto  $y$  ser satisfeita

Determinar o mercado  $j^*$ , tal que  $j^* = \arg \min_{j \in MNV} \{R(j, T_x) + F(j, T_x)\}$ .

Em que  $R(j, T_x)$  é o custo mínimo de viagem da rota  $T_x \cup \{j^*\}$  e  $F(j, T_x)$  é o custo mínimo total dos produtos comprados na rota  $T_x$ .

Para cada mercado  $j \in MNV$ ,  $R(j, T_x)$  é obtido através do algoritmo  $\text{custorota}(x, c, c', MV \cup \{j\})$ .  $F(j, T_x)$  é obtido através do algoritmo  $\text{custocompras}(x, a, pa, q, MV \cup \{j\})$ .

Atualizar a rota,  $T_x = T_x \cup \{j^*\}$ , e atualizar os mercados visitados e não visitados,  $MNV \leftarrow MNV \setminus \{j^*\}$ ,  $MV \leftarrow MV \cup \{j^*\}$

$y \leftarrow y + 1$ .

## Fase de Melhoramentos

### Passo 5 (*Market Drop*)

Remover um mercado da rota, caso se obtenha uma solução admissível e com um custo associado mais baixo que a anterior. Repetir, até não haver melhorias possíveis com este método.

(Repetir  $\forall u \in MV$ )

Usando o algoritmo  $\text{custocompras}(x, a, pa, q, MV \setminus \{u\})$ , verificar se é possível satisfazer a procura do cliente sem visitar o mercado  $u$ .

Se não for possível, saltar imediatamente para o próximo mercado.

Se for possível, usar o algoritmo  $\text{custorota}(x, c, c', MV \setminus \{u\})$  para obter o custo mínimo da rota  $T_x \setminus \{u\}$ .

Se se obtiver uma solução admissível com custo associado menor que a solução anterior, esta retém-se e repete-se o método depois de atualizar  $T_x \leftarrow T_x \setminus \{u\}$ ,  $MV \leftarrow MV \setminus \{u\}$  e  $MNV \leftarrow MNV \cup \{u\}$ .

Se não se obtiver uma solução melhor, saltar para o próximo método.

### Passo 6 (*Market Exchange*)

Substituir um dos mercados visitados por um não visitado. Reter qualquer solução possível que resulte numa redução do custo total. Repetir, até não existir nenhuma combinação de mercados visitado e não visitado que provoque melhoria no custo total segundo este método. Caso o conjunto de mercados não visitados seja vazio, saltar este método.

(Repetir  $\forall i \in MV, j \in MNV$ )

Usando o algoritmo  $\text{custocompras}(x, a, pa, q, MV \setminus \{i\} \cup \{j\})$ , verificar se é possível satisfazer a procura do cliente, trocando o mercado  $i$  pelo  $j$ .

Se não for possível, saltar imediatamente para o próximo conjunto de mercados visitado e não visitado.

Se for possível, usar o algoritmo  $\text{custorota}(x, c, c', MV \setminus \{i\} \cup \{j\})$  para obter o custo mínimo da rota  $T_x \setminus \{i\} \cup \{j\}$ .

Se se obtiver uma solução admissível com custo associado menor que a solução anterior, esta retém-se e repete-se o método, depois de atualizar  $T_x \leftarrow T_x \setminus \{i\} \cup \{j\}$ ,  $MV \leftarrow MV \setminus \{i\} \cup \{j\}$  e  $MNV \leftarrow MNV \setminus \{j\} \cup \{i\}$ .

Se não se obtiver uma solução com custo associado menor, saltar este método.

Depois dos melhoramentos, retira-se a cada mercado a quantidade comprada de cada produto:

$\forall i \in P, \forall j \in \{1, \dots, m\}: a_{i,j} \leftarrow a_{i,j} - f_{i,j,x}$ .

Passa-se ao próximo cliente e repete-se o procedimento.

$x \leftarrow x + 1$ .

### 3.3.2. ALGORITMO PARA A FUSÃO DE ROTAS

No algoritmo 3.2, apresenta-se o método utilizado para a fusão de rotas obtidas com o algoritmo 3.1. Juntam-se iterativamente os clientes, garantindo que não se excede a capacidade do veículo. Consideram-se os casos em que os clientes têm mercados em comum e em que os clientes não têm mercados em comum. Nas figuras 3.1., 3.2., 3.3. e 3.4. apresentam-se os dois casos possíveis para a fusão de rotas.

Cada cliente corresponde a uma rota que é constituída pelo cliente e os mercados que o servem. Para simplificar, ignora-se em cada rota o depósito.

#### Algoritmo 3.2

Reordenar o vetor dos clientes  $C := (1, 2, \dots, k)$ , aleatoriamente.

(Repetir enquanto  $\#(C) > 1$ )

$R := (C_1)$ . (primeiro elemento do vetor  $C$ )

$C \leftarrow C \setminus C_1$ .

$z := \#(C)$ . (cardinal do vetor de clientes)

$SF := \emptyset$ . (solução final com as rotas finais)

$x := 1$ .

(Repetir enquanto  $x \leq z$ )

Verificar se  $C_x$  tem mercados em comum com algum dos elementos de  $R$ .

Caso não tenha mercados em comum,  $R \leftarrow R * (C_x)$ ,  $C \leftarrow C \setminus C_x$ .

(\* representa concatenação de vetores)

Caso contrário, verificar com quais dos clientes em  $R$ ,  $C_x$  tem mercados em comum.

Não se podendo visitar cada mercado mais do que uma vez, averiguar, para todos os troços da rota, se a capacidade máxima do veículo,  $Q$ , não é excedida quando se adquirem os produtos para o cliente  $C_x$  nos mercados comuns. O mercado não é visitado uma segunda vez.

Se não é excedida,  $R \leftarrow R * (C_x)$ ,  $C \leftarrow C \setminus C_x$ .

$x \leftarrow x + 1$ .

Finaliza-se  $R$  com o depósito,  $O$ , no início e fim.

$SF \leftarrow SF \cup R$ .

Considere-se o caso em que não existem mercados em comum entre os clientes. Sejam  $(0, M_1, Cl_1, M_2, Cl_2, 0)$  e  $(0, M_3, Cl_3, 0)$  rotas pertencentes a uma solução admissível para o problema em estudo. Não existem mercados em comum nas duas rotas tal como se apresenta na figura 3.1.

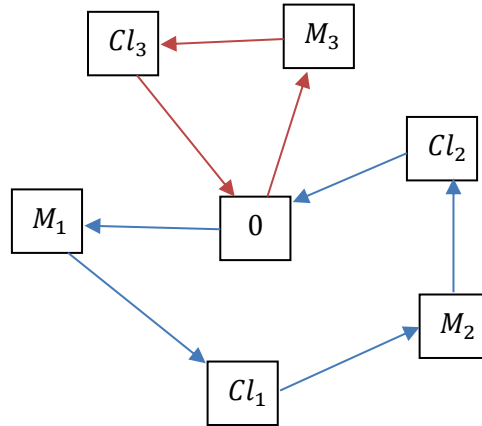


Figura 3.1: Exemplo de rotas sem mercados em comum.

Para juntar as rotas, neste caso não há problemas de exceder a capacidade do veículo. Já se tinha feito a junção dos clientes  $Cl_1$  e  $Cl_2$  numa rota e agora adiciona-se a essa rota o cliente  $Cl_3$ . Assim, removem-se as arestas  $(Cl_2, 0)$  e  $(0, M_3)$  e adiciona-se a aresta  $(Cl_2, M_3)$ . A rota final será:  $(0, M_1, Cl_1, M_2, Cl_2, M_3, Cl_3, 0)$ , tal como se mostra na figura 3.2.

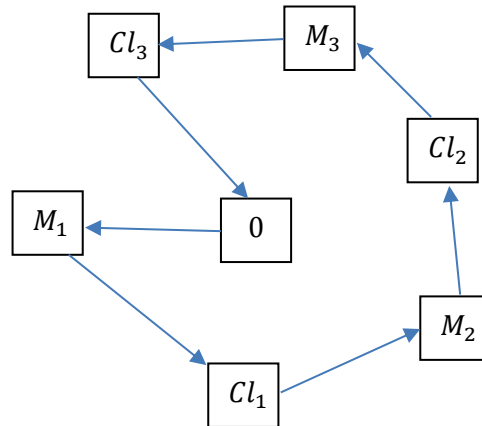


Figura 3.2: Fusão de rotas sem mercados em comum.

Note-se que ao juntar dois clientes numa só rota, tem-se sempre uma redução nos custos de viagem, visto que os custos de viagem respeitam a desigualdade triangular.

Considere-se o caso em que existem mercados em comum entre as rotas que se vão juntar. Sejam  $(0, M_1, M_2, Cl_1, M_3, M_4, Cl_2, 0)$  e  $(0, M_2, M_3, Cl_3, 0)$  rotas pertencentes a uma solução admissível para o problema em estudo. Os mercados  $M_2$  e  $M_3$  são comuns às duas rotas, como se apresenta na figura 3.3.

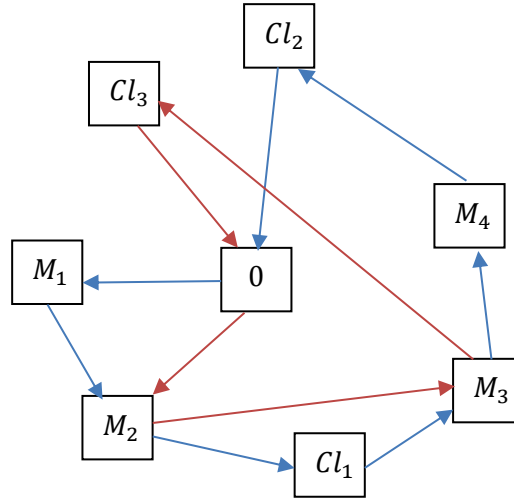


Figura 3.3: Exemplo de rotas com mercados em comum.

Para juntar as rotas, como não se pode visitar um mercado mais do que uma vez, quando se visitam  $M_2$  e  $M_3$  para comprar os produtos necessários para  $Cl_1$  e  $Cl_2$ , respetivamente, compram-se também os produtos para o cliente  $Cl_3$ . Contudo, tal só é possível se se verificar que não se excede a capacidade do veículo nos dois troços da rota,  $(M_1, M_2, Cl_1)$  e  $(M_3, M_4, Cl_2)$ .

A rota final será:  $(0, M_1, M_2, Cl_1, M_3, M_4, Cl_2, Cl_3, 0)$ . Como se pode ver na figura 3.4.

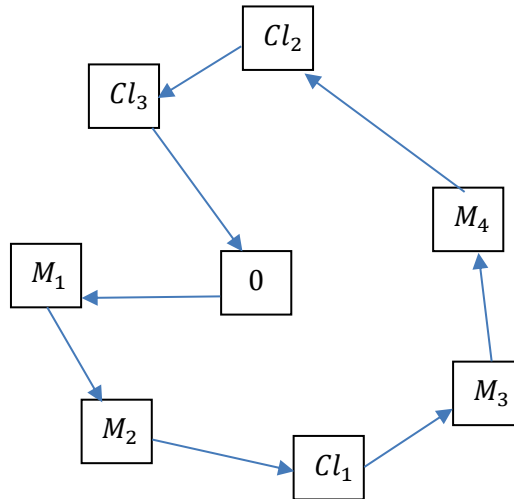


Figura 3.4: Fusão de rotas com mercados em comum

Na próxima secção apresentam-se os fluxogramas correspondentes às funções custorota (figura 3.5.) e custocompras (figura 3.6.).



### 3.4. FLUXOGRAMAS

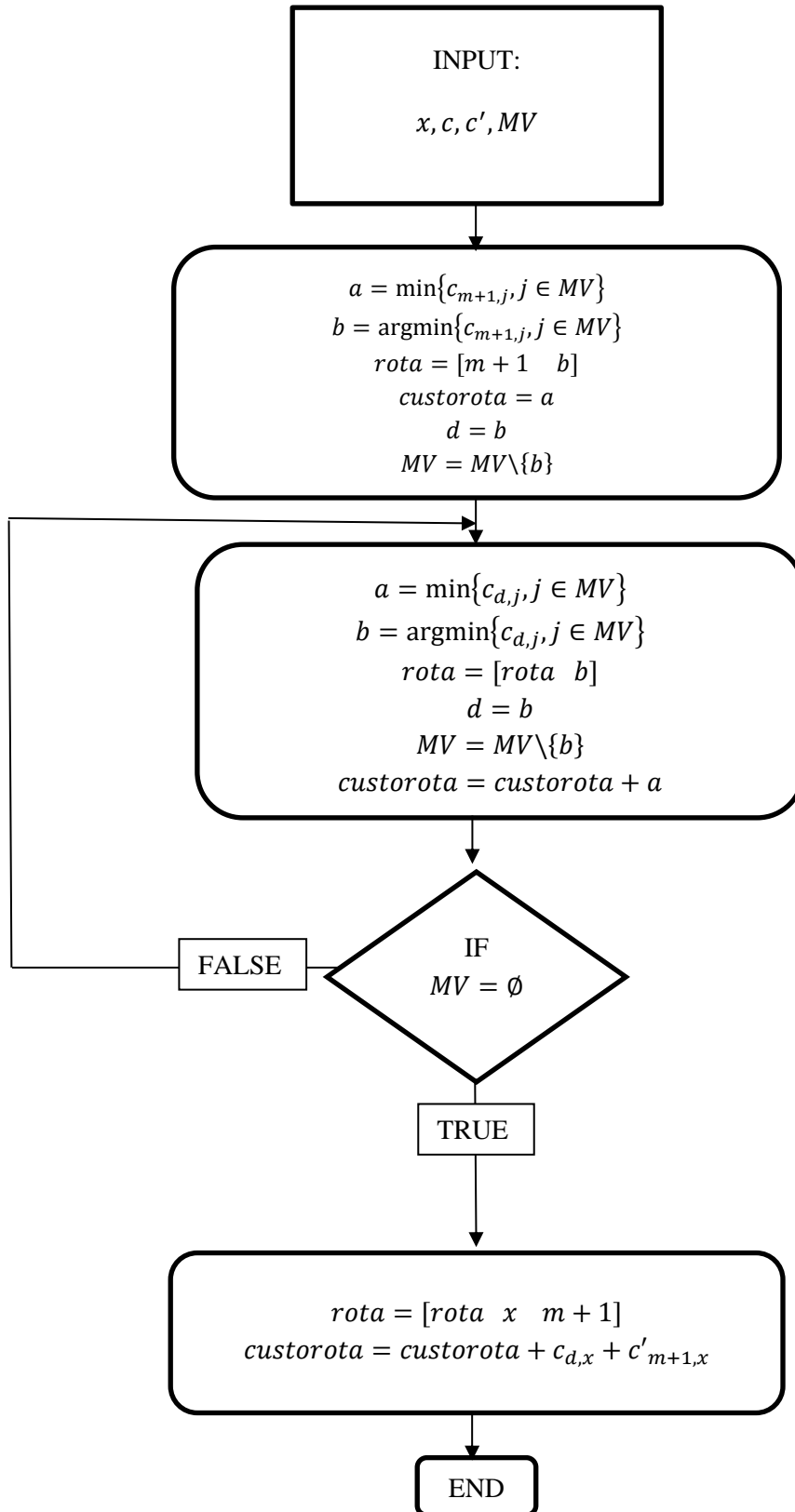


Figura 3.5: Fluxograma da função custorota.

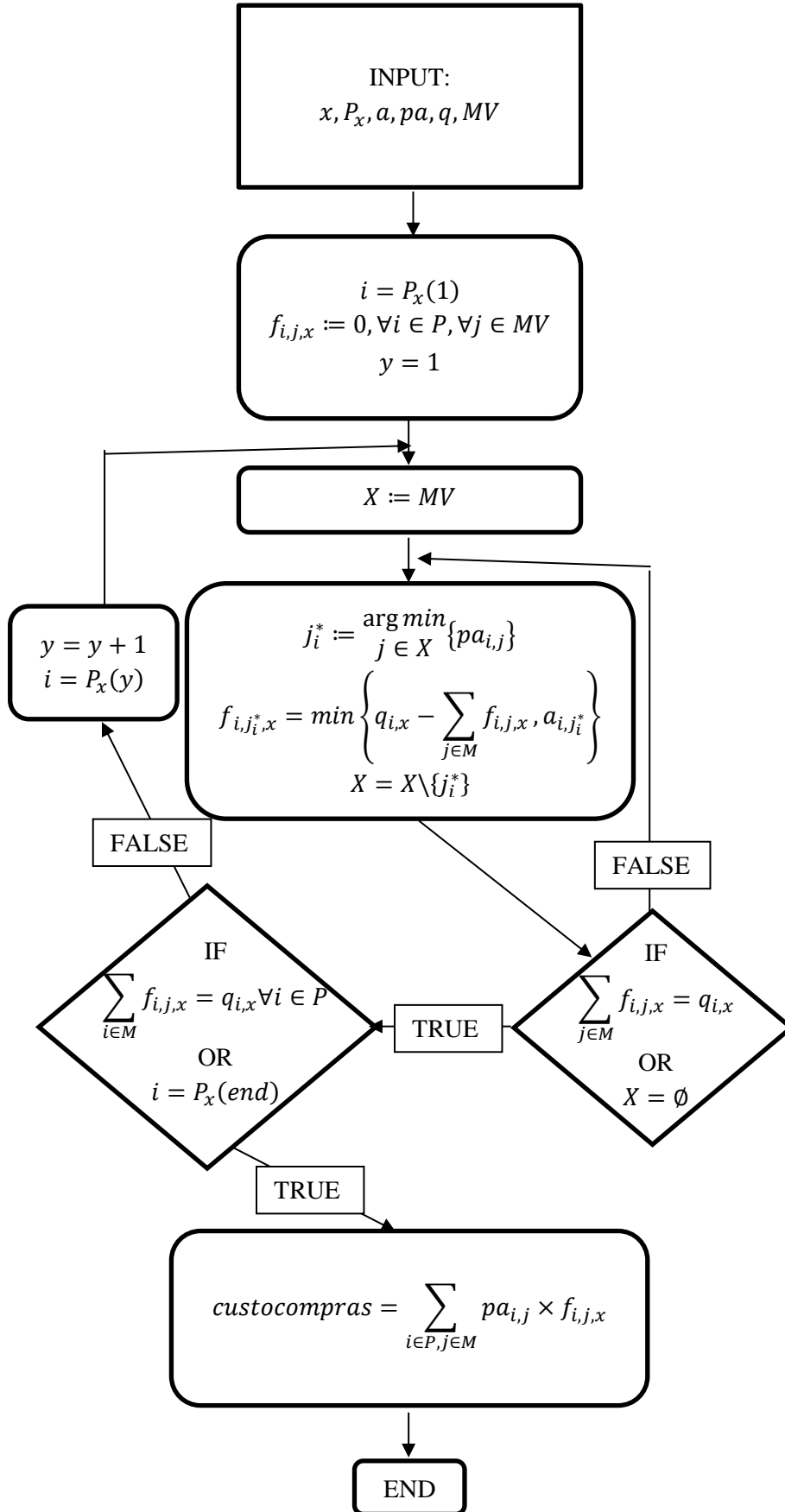


Figura 3.6: Fluxograma da função custocompras.

## 4. RESULTADOS COMPUTACIONAIS

Neste capítulo, apresentam-se os resultados computacionais obtidos para o problema de determinação de rotas de recolha e distribuição de produtos com escolha seletiva de mercados, considerando dados gerados aleatoriamente. Apresenta-se ainda um exemplo ilustrativo da aplicação da heurística implementada.

Os algoritmos apresentados no capítulo anterior para a obtenção de soluções admissíveis para este problema foram implementados em MATLAB.

Geraram-se vários grupos de dados correspondentes a diferentes valores dos parâmetros:

- área de localização de depósito, clientes e mercados;
- probabilidade associada à disponibilidade de um produto num mercado e probabilidade associada à procura de um produto por parte de cada cliente;
- número de clientes e mercados;
- número de produtos;
- intervalo para a procura dos clientes;
- intervalo para a disponibilidade dos produtos nos mercados;
- intervalo para os custos de aquisição dos produtos nos mercados;
- densidade da matriz das distâncias;
- capacidade dos veículos.

Em cada uma das duas áreas consideradas, geraram-se as coordenadas dos clientes, depósito e mercados. Construiu-se a matriz das distâncias, calculando as distâncias euclidianas entre depósito, mercados e clientes, e considerando uma probabilidade de existência de um caminho direto entre estes elementos. Determinou-se o caminho mais curto entre qualquer par de pontos, recorrendo ao algoritmo de Floyd, descrito em [20]. Garante-se, assim, que as distâncias verificam a desigualdade triangular.

Todos os testes foram efetuados num computador com o processador AMD Ryzen 5 1500X à frequência de 3.5GHz com 16GB de RAM.

Comum a todos os grupos de dados:

- número de produtos, 4;
- intervalo da procura para cada cliente, [8,30];
- intervalo da disponibilidade de cada produto em cada mercado, [5, 50];
- intervalo dos custos de aquisição unitários de cada produto em cada um dos mercados em que estão disponíveis, [5,15];
- densidade da matriz das distâncias, 80% (ou seja, a probabilidade de haver um caminho direto de um vértice para outro).

Tabela 4.1: Grupos de dados gerados

Área	Grupo	Probabilidade de um mercado ter um produto	Probabilidade de um cliente necessitar de um produto	Número de clientes	Número de mercados
$[0,90] \times [0,60]$	Grupo 1	70%	70%	5	10
				10	12
				15	18
	Grupo 2	80%	60%	5	10
				10	12
				15	18
$[0,120] \times [0,120]$	Grupo 3	70%	70%	5	10
				10	12
				15	18
	Grupo 4	80%	60%	5	10
				10	12
				15	18

Todos os dados são testados com uma capacidade máxima para cada veículo de 150 e 250. Considerou-se a capacidade de 150 por corresponder, aproximadamente, ao dobro da maior procura por parte dos clientes, quando a probabilidade de haver procura de um produto é de 70%. A capacidade de 250 foi considerada de modo a permitir a existência de rotas que servem um maior número de clientes.

No algoritmo usado para a determinação das rotas para o problema em estudo, na fase de determinação de rotas com um cliente, ordenaram-se aleatoriamente os clientes para atribuição dos mercados que os servem. Para determinar o número de iterações para cada grupo de dados gerados, procedeu-se do modo que se descreve seguidamente.

Consideraram-se quatro exemplos para cada par número de clientes, número de mercados, que diferem em outros parâmetros. Para cada exemplo, executa-se dez vezes o programa para cada número de iterações de 10, 15 e 20. Calculou-se a média dos custos finais para cada um dos exemplos correspondente a cada número de iterações e compararam-se os ganhos percentuais das médias com cada aumento do número de iterações.

Os ganhos percentuais em termos de redução dos custos finais de 10 iterações para 15 iterações e de 15 para 20 são de menos de 1%, para os exemplos de dados com 5 clientes e 10 mercados e para os exemplos com 15 clientes e 18 mercados. Os ganhos percentuais entre números de iterações, para os exemplos de dados com 10 clientes e 12 mercados, são de menos de 2%.

Como a redução dos custos finais é muito pequena, não se justifica o aumento do tempo computacional de entre 20% e 50% ao executar o programa com mais iterações. O número de iterações para cada exemplo de dados ficou então 10, visto que, em média, se consegue um resultado com um custo final que se pensa ser próximo do melhor valor que se consegue obter com este algoritmo.

## 4.1. EXEMPLO DA FORMAÇÃO DE ROTAS

Nesta secção apresenta-se um exemplo com dados respeitantes a um exemplo do Grupo 1 com 5 clientes e 10 mercados e com uma capacidade dos veículos de 150.

Sejam as matrizes de procura de cada cliente, de disponibilidade e de preço de aquisição unitário de cada produto em cada mercado, respetivamente:

$$\begin{pmatrix} 0 & 0 & 23 & 0 \\ 0 & 30 & 27 & 16 \\ 16 & 0 & 0 & 15 \\ 26 & 0 & 0 & 12 \\ 16 & 9 & 25 & 0 \end{pmatrix}, \begin{pmatrix} 40 & 0 & 33 & 0 \\ 0 & 39 & 0 & 0 \\ 0 & 28 & 0 & 22 \\ 9 & 11 & 0 & 39 \\ 38 & 32 & 5 & 0 \\ 6 & 28 & 10 & 46 \\ 43 & 39 & 49 & 17 \\ 16 & 0 & 15 & 0 \\ 0 & 37 & 0 & 16 \\ 0 & 22 & 21 & 0 \end{pmatrix} \text{ e } \begin{pmatrix} 14 & 0 & 9 & 0 \\ 0 & 9 & 0 & 0 \\ 0 & 15 & 0 & 11 \\ 10 & 11 & 0 & 5 \\ 14 & 11 & 6 & 0 \\ 11 & 12 & 13 & 10 \\ 14 & 7 & 6 & 7 \\ 10 & 0 & 13 & 0 \\ 0 & 12 & 0 & 7 \\ 0 & 14 & 9 & 0 \end{pmatrix}.$$

Sejam as matrizes de custos de viagens entre mercados/depósito e mercados/depósito, entre mercados/depósito e clientes e entre clientes e clientes, respetivamente:

$$\begin{pmatrix} 0 & 53 & 8 & 77 & 63 & 68 & 63 & 47 & 35 & 31 & 33 \\ 53 & 0 & 56 & 26 & 21 & 38 & 19 & 33 & 30 & 52 & 21 \\ 8 & 56 & 0 & 80 & 68 & 60 & 65 & 39 & 27 & 39 & 38 \\ 77 & 26 & 80 & 0 & 34 & 39 & 34 & 47 & 53 & 69 & 47 \\ 63 & 21 & 68 & 34 & 0 & 59 & 40 & 54 & 43 & 70 & 32 \\ 68 & 38 & 60 & 39 & 59 & 0 & 57 & 21 & 43 & 39 & 45 \\ 63 & 19 & 65 & 34 & 40 & 57 & 0 & 51 & 43 & 70 & 30 \\ 47 & 33 & 39 & 47 & 54 & 21 & 51 & 0 & 22 & 23 & 30 \\ 35 & 30 & 27 & 53 & 43 & 43 & 43 & 22 & 0 & 29 & 13 \\ 31 & 52 & 39 & 69 & 70 & 39 & 70 & 23 & 29 & 0 & 40 \\ 33 & 21 & 38 & 47 & 32 & 45 & 30 & 30 & 13 & 40 & 0 \end{pmatrix},$$

$$\begin{pmatrix} 55 & 56 & 80 & 36 & 17 \\ 43 & 77 & 28 & 42 & 38 \\ 60 & 58 & 84 & 35 & 19 \\ 69 & 100 & 54 & 68 & 61 \\ 54 & 88 & 49 & 53 & 51 \\ 67 & 101 & 66 & 66 & 64 \\ 52 & 86 & 47 & 51 & 49 \\ 52 & 86 & 61 & 51 & 49 \\ 35 & 69 & 58 & 34 & 32 \\ 62 & 87 & 80 & 61 & 48 \\ 22 & 56 & 48 & 21 & 19 \end{pmatrix} \text{ e } \begin{pmatrix} 0 & 77 & 34 & 43 & 41 \\ 77 & 0 & 92 & 40 & 39 \\ 34 & 92 & 0 & 53 & 65 \\ 43 & 40 & 53 & 0 & 19 \\ 41 & 39 & 64 & 19 & 0 \end{pmatrix}.$$

Os gráficos abaixo representam as coordenadas dos mercados, clientes e do depósito e as rotas obtidas ao executar os algoritmos.

Obtiveram-se as rotas  $(O, M_1, C_1, O)$ ,  $(O, M_7, C_2, O)$ ,  $(O, M_7, M_4, C_3, O)$ ,  $(O, M_9, M_1, C_4, O)$  e  $(O, M_7, M_8, M_1, C_5, O)$  com o algoritmo de obtenção de rotas com um só cliente.

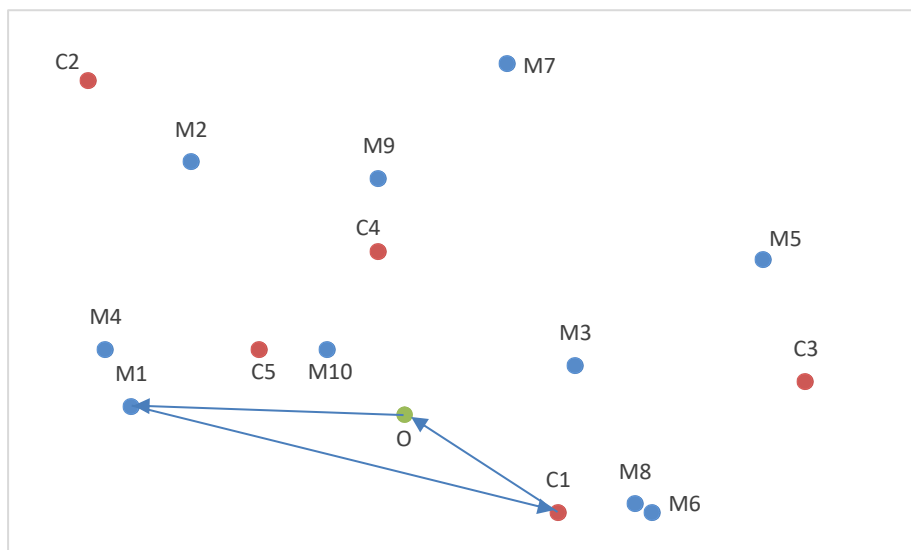


Figura 4.1: Rota  $(O, M_1, C_1, O)$

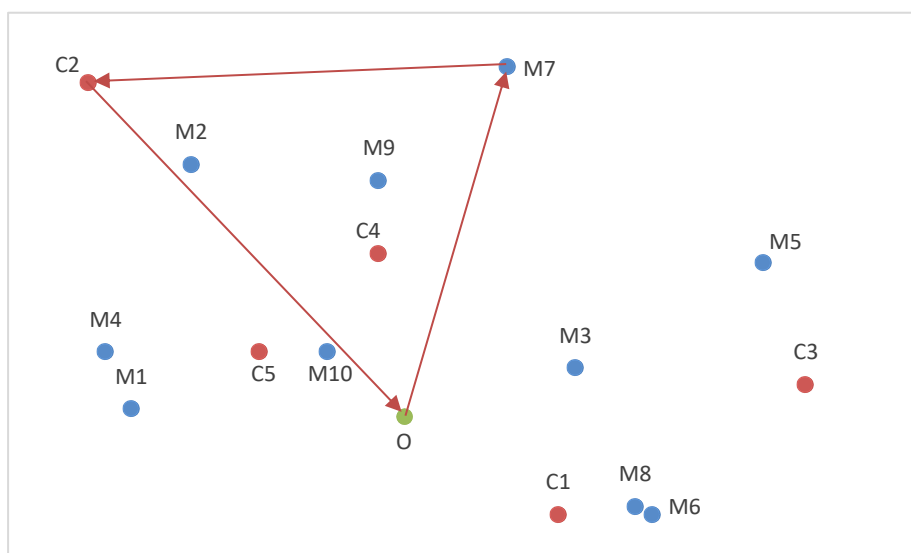


Figura 4.2: Rota  $(O, M_7, C_2, O)$

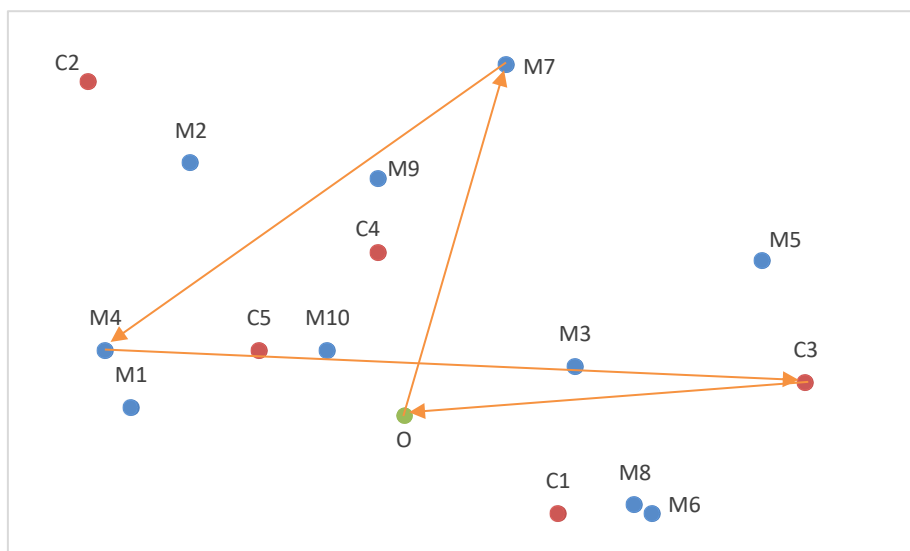


Figura 4.3: Rota  $(O, M_7, M_4, C_3, O)$

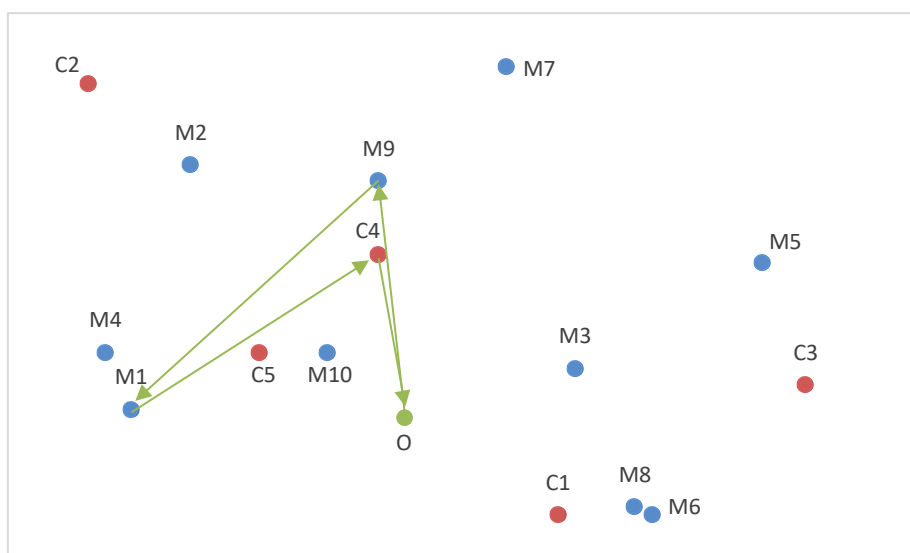


Figura 4.4: Rota  $(O, M_9, M_1, C_4, O)$

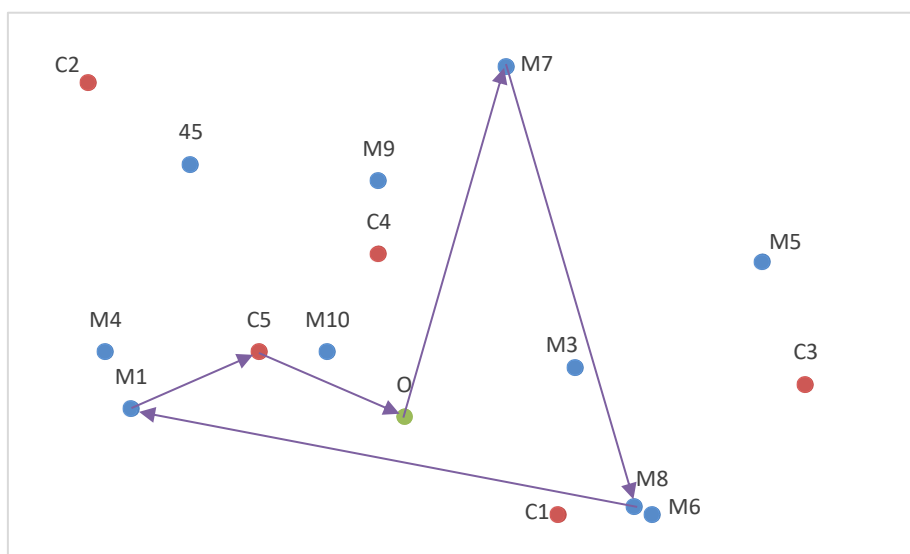


Figura 4.5: Rota  $(O, M_7, M_8, M_1, C_5, O)$

Obteve-se a rota  $(O, M_7, C_1, C_5, M_8, C_4, M_4, C_3, M_1, C_2, O)$  com o algoritmo de obtenção de rotas com um só cliente seguido pelo algoritmo de fusão de rotas. Note-se que não é a aplicação do segundo algoritmo à solução anterior. Os algoritmos foram executados desde o início. Neste exemplo, uma capacidade de 250 resultaria num custo total semelhante, visto que, com uma capacidade de 150 já é possível juntar todas as rotas numa só.

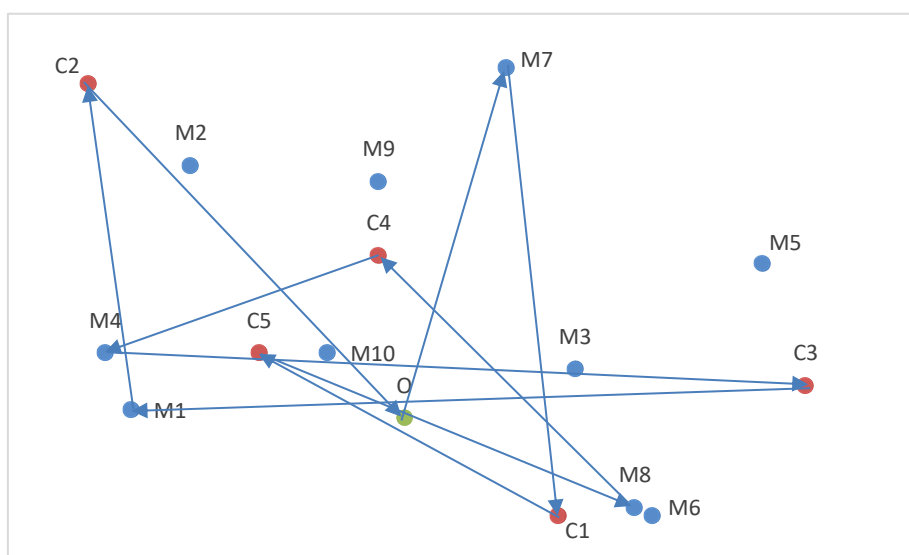


Figura 4.6: Rota  $(O, M_7, C_1, C_5, M_8, C_4, M_4, C_3, M_1, C_2, O)$

## 4.2. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Nesta secção, apresentam-se os resultados obtidos com os dados gerados aleatoriamente e faz-se uma análise dos resultados.

Na tabela 4.2, apresentam-se as comparações dos custos finais que se obtiveram entre executar o programa com os dois algoritmos em sequência, isto é, o algoritmo de obtenção de rotas com um só cliente a que se segue sequencialmente a fusão de rotas e executar o programa só com o primeiro algoritmo, isto é, o algoritmo de obtenção de rotas só com um cliente.

Nesta tabela, Média 1 é a melhoria percentual que o algoritmo de fusão de rotas provocou, em média, no custo final para cada classe de exemplos com uma capacidade de veículos de 150. Desvio padrão 1 é o desvio padrão da amostra da melhoria percentual referida anteriormente.

Média 2 é a melhoria, em média, no custo final para cada classe de exemplos com uma capacidade de veículos de 250, comparada com a anterior de 150. Desvio padrão 2 é o desvio padrão da amostra da melhoria percentual referida anteriormente.



Tabela 4.2: Comparação dos custos finais

Grupo	Número de clientes	Número de mercados	Média 1	Desvio padrão 1	Média 2	Desvio padrão 2
Grupo 1	5	10	13.67%	5.36%	2.65%	5.38%
	10	12	11.68%	7.71%	4.59%	2.61%
	15	18	9.88%	1.45%	1.89%	1.14%
Grupo 2	5	10	10.54%	4.00%	5.60%	4.70%
	10	12	16.40%	2.16%	2.14%	1.95%
	15	18	11.66%	1.26%	1.64%	2.22%
Grupo 3	5	10	15.40%	3.61%	2.06%	3.90%
	10	12	13.65%	1.19%	4.60%	6.06%
	15	18	12.83%	3.32%	3.42%	2.23%
Grupo 4	5	10	16.12%	3.79%	2.79%	3.96%
	10	12	15.01%	2.20%	6.52%	0.84%
	15	18	19.39%	6.39%	2.15%	1.15%

A junção de rotas, uma vez que os custos de viagem são proporcionais às distâncias e estas verificam a desigualdade triangular, resulta sempre numa redução do custo final, tal como se apresenta na tabela 4.2 (ver Média 1).

Verifica-se que, tal como seria de esperar, a junção de mais rotas, devido à maior capacidade do veículo, provoca uma redução no custo (ver Média 2 na tabela 4.2.). Esta redução seria maior se se considerassem no problema custos fixos de utilização de cada veículo. Ao juntar rotas, o número de veículos usados diminuiria e os custos fixos também.

Na tabela 4.3, apresentam-se as médias percentuais de melhoria quando se agrupam os dados com a mesma probabilidade de procura dos clientes, isto é, os Grupos 1 e 3, onde a probabilidade de procura é de 70% e os Grupos 2 e 4, onde a probabilidade de procura é de 60%.

Tabela 4.3: Média das melhorias por probabilidade da procura

Grupos	Capacidade 150	Capacidade 250
1 e 3	12.85%	3.20%
2 e 4	13.18%	3.47%

Verifica-se que a média das melhorias percentuais é maior (embora com pouca diferença) para os Grupos 2 e 4, onde a probabilidade de um cliente ter procura de um produto diferente de zero é menor. Tal deve-se ao facto de menores quantidades serem transportadas para cada cliente, o que facilita a sua junção numa rota com outros clientes.

Na tabela 4.4, apresentam-se as médias percentuais de melhoria quando se agrupam os dados com a mesma área, isto é, Grupos 1 e 2 (área  $[0,90] \times [0,60]$ ) e Grupos 3 e 4 (área  $[0,120] \times [0,120]$ ).

Tabela 4.4: Média das melhorias por áreas

Grupos	Capacidade 150	Capacidade 250
1 e 2	12.30%	3.09%
3 e 4	15.40%	3.59%

Verifica-se que a média das melhorias percentuais é maior para os Grupos 3 e 4, cuja área é maior. Tal deve-se ao facto de as distâncias percorridas serem maiores, o que resulta numa maior redução quando se faz a junção de rotas.

Na tabela 4.5, apresentam-se as médias percentuais de melhoria quando se agrupam os dados para cada par (número de clientes/número de mercados).

Tabela 4.5: Média das melhorias por par cliente/mercado

Pares	Capacidade 150	Capacidade 250
(5,10)	13.93%	3.28%
(10,12)	14.19%	4.46%
(15,18)	13.44%	2.28%

Verifica-se que as melhores médias dizem respeito aos casos de 10 clientes e 12 mercados. Eventualmente, maiores capacidades do veículo seriam mais favoráveis aos casos com maior número de clientes.

Na tabela 4.6, apresentam-se as comparações dos tempos computacionais que se obtiveram entre executar o programa com os dois algoritmos e executar o programa só com o primeiro algoritmo.

Média 3 é o aumento percentual em tempo computacional que o algoritmo de fusão de rotas provocou, em média, para cada classe de exemplos com uma capacidade de veículos de 150. Desvio padrão 3 é o desvio padrão da amostra do aumento percentual referido anteriormente.

Média 4 é o aumento, em média, no tempo computacional para cada classe de exemplos com uma capacidade de veículos de 250, comparada com a anterior de 150. Desvio padrão 4 é o desvio padrão do aumento percentual referido anteriormente.

Tabela 4.6: Comparação dos tempos computacionais

Grupo	Número de clientes	Número de mercados	Média 3	Desvio padrão 3	Média 4	Desvio padrão 4
Grupo 1	5	10	6.15%	16.62%	10.32%	19.02%
	10	12	3.01%	10.11%	-2.14%	9.04%
	15	18	7.58%	3.99%	-3.63%	5.22%
Grupo 2	5	10	2.36%	20.16%	13.96%	22.58%
	10	12	1.72%	10.09%	-3.83%	4.00%
	15	18	13.56%	7.12%	-2.33%	3.45%
Grupo 3	5	10	-0.37%	16.73%	3.25%	16.71%
	10	12	8.16%	10.23%	2.96%	12.69%
	15	18	1.84%	7.91%	-15.95%	48.93%
Grupo 4	5	10	2.13%	10.28%	0.14%	18.78%
	10	12	-2.12%	16.81%	2.60%	23.93%
	15	18	6.60%	5.05%	-2.19%	14.11%

Observando a tabela 4.6, podemos concluir que na maioria dos casos, considerando a capacidade do veículo igual a 150 (ver Média 3), o tempo computacional é maior quando se executam sequencialmente os dois algoritmos, como seria de esperar. As exceções, tendo em conta as médias, são apenas o par clientes/mercados (5,10) do Grupo 3 e (10,12) do Grupo 4. Quando se considera uma capacidade do veículo de 250, só em 50% dos casos se verifica que o tempo computacional médio é superior ao correspondente a uma capacidade de 150.

Há uma grande variabilidade nos tempos computacionais (ver Desvio padrão 3 e Desvio padrão 4).

## 5. CONCLUSÕES

Neste trabalho de projeto, apresentou-se uma heurística para um problema de recolha e distribuição de produtos com escolha seletiva de mercados, isto é, um problema em que se pretende determinar rotas para vários veículos de modo a satisfazer a procura de um conjunto de clientes a partir de mercados que praticam preços distintos para os vários produtos. Considerou-se uma frota homogênea de veículos que partem do depósito, visitam alguns mercados para adquirir os produtos de modo a satisfazer a procura dos clientes, distribuem os produtos pelos clientes e regressam ao depósito. Neste problema, contrariamente a outros problemas de determinação de rotas, não é obrigatório visitar todos os mercados. O objetivo consiste em determinar rotas que verifiquem estas condições com custo total mínimo. O custo tem duas componentes: custo de viagem, proporcional à distância percorrida pelos veículos e custo de aquisição dos produtos nos mercados.

Este problema aplica-se a várias situações reais tais como o serviço de partilha de bicicletas. Os veículos partem de um depósito e, a determinadas horas do dia, vão a alguns dos pontos de partilha para recolher as bicicletas e transportá-las para os locais onde há maior procura nessa hora. Visitar só alguns dos pontos de partilha em vez de todos tem um custo associado menor. Outro exemplo é o caso de algumas empresas que prestam serviços a outras e precisam de adquirir algum do material necessário em mercados, antes e a caminho das empresas onde vão prestar o serviço.

O problema em estudo é semelhante ao bem conhecido TPP, embora no caso do TPP só se considere um veículo e este parte e chega ao depósito depois de adquirir produtos em vários mercados sem os distribuir por clientes. Também, neste problema não é necessário visitar todos os mercados. Esta semelhança levou ao desenvolvimento de uma heurística em que, numa primeira fase, se constroem rotas para um só veículo com base em heurísticas já existentes para o TPP. Estas rotas incluem apenas um cliente e os mercados que o servem, além do depósito. Numa segunda fase, faz-se a junção destas rotas com um só cliente, enquanto existir capacidade no veículo.

A heurística apresentada neste trabalho de projeto foi implementada em MATLAB. Foi testada a heurística completa, incluindo a fusão de rotas com duas capacidades do veículo diferentes versus apenas o primeiro algoritmo de obtenção de rotas só com um cliente, para verificar a redução do custo final resultante da fusão.

Os resultados computacionais foram obtidos com dados gerados aleatoriamente com diferentes valores para certos parâmetros. Para analisar os resultados, estes agruparam-se de acordo com os parâmetros mais importantes: a probabilidade da procura para cada produto de cada cliente ser diferente de zero, a área onde estão o depósito, os mercados e os clientes, o número de clientes e de mercados. Concluiu-se que a fusão de rotas provocou maior melhoria nos custos finais nos exemplos em que se considerou uma área maior.

Verificou-se que os tempos computacionais não aumentam muito com a fusão de rotas em média, quando comparado com a obtenção das rotas com um só cliente, logo a fase de fusão de rotas foi implementada eficientemente.

Como trabalho futuro, a heurística desenvolvida neste projeto poderia ser melhorada com a ajuda de algoritmos de pesquisa local depois de se fazer a fusão de rotas. Trocar a ordem pela qual algum cliente e algum mercado são visitados, se tal for possível, pode provocar uma redução no custo, resultando numa melhor solução. Também haveria interesse em testar os resultados desta heurística contra a solução ótima, de modo a verificar se, com a heurística, é possível obter soluções próximas da ótima. Tal seria possível apenas para exemplos com pequenas dimensões, devido à complexidade computacional elevada dos algoritmos exatos para o problema.

Se se considerasse o mesmo problema com a variante de existir um custo fixo associado a utilizar um veículo, por exemplo, por ter de se contratar mais um funcionário, e adaptando o modelo e a heurística para acomodar esta nova restrição, os resultados iriam, em princípio, ser ainda mais favoráveis para a fusão de rotas.

# REFERÊNCIAS

- [1] G. Desaulniers, O. B. Madsen e S. Ropke, “The Vehicle Routing Problem with Time Windows,” em *Vehicle Routing: Problems, Methods and Applications*, P. Toth e D. Vigo, Edits., 2014, pp. 119-159.
- [2] M. Battarra, J.-F. Cordeau e M. Iori, “Pickup-and-Delivery Problems for Goods Transportation,” em *Vehicle Routing: Problems, Methods and Applications*, P. Toth e D. Vigo, Edits., 2014, pp. 161-191.
- [3] D. Manerba, R. Mansini e J. Riera-Ledesma, “The Traveling Purchaser Problem and its Variants,” *European Journal of Operational Research*, vol. 259, nº 1, pp. 1-18, 2017.
- [4] C.-K. Ting, X.-L. Liao, Y.-H. Huang e R.-T. Liaw, “Multi-vehicle selective pickup and delivery using metaheuristic algorithms,” *Information Sciences*, Vols. %1 de %2406-407, pp. 146-169, 2017.
- [5] C. Archetti, M. Speranza e D. Vigo, “Vehicle Routing Problems with Profits,” em *Vehicle Routing: Problems, Methods and Applications*, P. Toth e D. Vigo, Edits., 2014, pp. 273-297.
- [6] P. J. Palomo-Martínez e M. A. Salazar-Aguilar, “The bi-objective traveling purchaser problem with deliveries,” *European Journal of Operational Research*, vol. 273, nº 2, pp. 608-622, 2019.
- [7] M. W. P. Savelsbergh e M. Sol, “The General Pickup and Delivery Problem,” *Transportation Science*, vol. 29, nº 1, pp. 1-105, 1995.
- [8] B. M. C. d. Oliveira, *Planning the distribution of agricultural products in a short distribution channel*, Faculdade de Ciências da Universidade de Lisboa - Dissertações de Mestrado, 2013.
- [9] P. Toth e D. Vigo, Edits., *The Vehicle Routing Problem*, Philadelphia: Society for Industrial and Applied Mathematics, 2002.
- [10] T. Öncan, I. K. Altinel e G. Laporte, “A comparative analysis of several asymmetric traveling salesman problem formulations,” *Computers & Operations Research*, vol. 36, nº 3, pp. 637-654, 2009.
- [11] S. C. Sarin, H. D. Sherali e A. Bhootra, “New tighter polynomial length formulations for the asymmetric traveling salesman problem with and without precedence constraints,” *Operations Research Letters*, vol. 33, nº 1, pp. 62-70, 2005.
- [12] A. Rais, F. Alvelos e M. S. Carvalho, “New mixed integer-programming model for the pickup-and-delivery,” *European Journal of Operational Research*, vol. 235, nº 3, pp. 530-539, 2014.
- [13] F. F. Bector, G. Laporte e J. Renaud, “Heuristics for the traveling purchaser problem,” *Computers & Operations Research*, vol. 30, nº 4, pp. 491-504, 2003.
- [14] T. Ramesh, “Traveling purchaser problem,” *OPSEARCH*, vol. 18, pp. 78-91, 1981.
- [15] H. L. Ong, “Approximate algorithms for the traveling purchaser problem,” *Operations Research Letters*, vol. 1, nº 5, pp. 201-205, 1982.

- [16] K. N. Singh e D. L. v. Oudheusden, “A Branch and Bound Algorithm fir the Traveling Purchaser Problem,” *European Journal of Operational Research*, vol. 97, pp. 571-579, 1997.
- [17] G. Laporte, J. Riera-Ledesma e J.-J. Salazar-González, “A Branch-and-Cut Algorithm for the Undirected Traveling Purchaser Problem,” *Networks*, vol. 32, n° 4, pp. 263-273, 1998.
- [18] M. J. Choi e S. H. Lee, “The multiple traveling purchaser problem for maximizing system's reliability with budget constraints,” *Expert Systems with Applications*, vol. 38, n° 8, pp. 9848-9853, 2011.
- [19] R. Bernardino e A. Paias, “Metaheuristics based on decision hierarchies for the traveling,” *International Transactions in Operational Research*, pp. 1-27, 2016.
- [20] H. A. Taha, *Operations Research. An Introduction*, Prentice Hall, 2017.

# ANEXO

Nas tabelas A.1 e A.2, apresentam-se os custos finais e tempos computacionais, em segundos, obtidos para todos os Grupos de dados gerados de acordo com os parâmetros apresentados no capítulo 4.

Tabela A.1: Resultados finais dos Grupos 1 e 2

			Algoritmo 1		Algoritmo 1 e 2, capacidade 150		Algoritmo 1 e 2, capacidade 250	
			Custo final	Tempo computacional (s)	Custo final	Tempo computacional (s)	Custo final	Tempo computacional (s)
Grupo 1	(5,10)	1	3303	34.29	2693	37.50	2682	39.60
		2	2496	13.92	2092	15.95	2122	16.78
		3	2555	19.39	2200	15.88	2177	21.83
		4	3167	29.31	2973	34.80	2659	32.37
	(10,12)	1	7019	50.84	6537	56.48	6109	50.54
		2	4440	50.06	3426	48.99	3373	44.83
		3	5696	43.72	5080	48.88	4725	53.20
		4	6306	74.61	5914	68.11	5719	69.20
	(15,18)	1	9472	135.30	8406	142.25	8147	141.11
		2	9189	134.00	8429	139.36	8387	132.37
		3	10108	167.79	9000	181.69	8865	184.61
		4	8261	143.32	7514	161.83	7327	145.17
Grupo 2	(5,10)	1	1897	2.89	1619	3.07	1578	3.78
		2	3249	23.21	3081	24.47	2703	22.28
		3	2179	12.97	1954	9.69	1846	13.68
		4	3083	14.56	2713	17.92	2657	18.01
	(10,12)	1	2997	18.52	2452	20.51	2407	18.98
		2	4439	29.73	3686	30.21	3632	30.12
		3	3190	13.99	2642	12.26	2633	12.21
		4	5635	42.14	4888	45.05	4648	41.84
	(15,18)	1	7723	107.54	6958	132.37	6623	122.78
		2	8206	93.29	7167	101.98	7173	99.75
		3	7526	84.94	6588	97.46	6568	97.79
		4	9437	169.60	8342	181.65	8214	181.23

Tabela A.2: Resultados finais dos Grupos 3 e 4

			Algoritmo 1		Algoritmo 1 e 2, capacidade 150		Algoritmo 1 e 2, capacidade 250	
			Custo final	Tempo computacional (s)	Custo final	Tempo computacional (s)	Custo final	Tempo computacional (s)
Grupo 3	(5,10)	1	2696	9.05	2174	10.85	2195	9.92
		2	3769	19.39	3244	20.50	3216	17.70
		3	3905	18.85	3470	17.20	3200	20.30
		4	3544	20.57	2936	16.80	2920	19.69
	(10,12)	1	6678	63.93	5724	63.47	5881	66.22
		2	6667	45.27	5865	46.56	5746	49.35
		3	7071	44.75	6112	54.84	5563	46.91
		4	6854	58.57	5844	63.23	5253	73.32
	(15,18)	1	10260	106.61	8594	120.97	8238	127.59
		2	10733	154.98	9769	155.19	9751	163.35
		3	9972	248.63	8850	239.84	8384	26.23
		4	11672	237.57	9937	231.16	9530	264.76
Grupo 4	(5,10)	1	2699	11.85	2322	12.55	2144	15.58
		2	4181	20.75	3504	18.51	3469	17.16
		3	2778	12.71	2183	12.69	2216	13.18
		4	2545	7.38	2217	8.38	2128	6.69
	(10,12)	1	5750	28.34	5004	28.97	4631	28.97
		2	4874	24.97	4022	20.03	3786	27.30
		3	5552	38.45	4651	45.82	4326	36.77
		4	6095	34.12	5281	30.68	4977	28.79
	(15,18)	1	7578	64.45	6256	64.51	6039	62.89
		2	9260	105.77	7600	117.17	7550	101.59
		3	9132	147.82	7884	155.35	7701	183.04
		4	7596	50.14	5428	55.36	5311	49.40